

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 04-195436

(43)Date of publication of application : 15.07.1992

(51)Int.Cl.

G06F 11/28

(21)Application number : 02-322955

(71)Applicant : HITACHI LTD

(22)Date of filing : 28.11.1990

(72)Inventor : HASHIYA TOSHIRO
KODAMA YASUTAKA

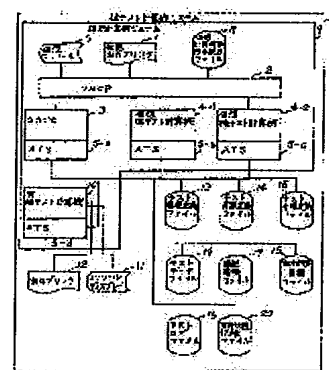
(54) AUTOMATIC TEST SYSTEM FOR COMPUTER SYSTEM

(57)Abstract:

PURPOSE: To simultaneously execute the tests set by plural persons in charge of test by giving plural procedures related to test circumstances, those related to test resources, and those related to test procedures independently of one another.

CONSTITUTION: A means is provided which is allowed to exist as test circumstances in the case of not only an actual computer 10 but also virtual computers 4-1 and 4-2 as the computer to be tested and executes the automatic test in either case as well as both cases. Test control procedures which prescribe the operation condition required for execution of the test in accordance with contents of the computer to be tested and test execution resource procedures which prescribe a test job to be used for the test and the use of data like commands are made independent of each other.

Thus, plural persons in charge of test can independently generate procedures to improve the productivity of test items.



⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A)

平4-195436

⑬ Int.Cl.⁵
G 06 F 11/28

識別記号 庁内整理番号
3 4 0 Z 7165-5B

⑭ 公開 平成4年(1992)7月15日

審査請求 未請求 請求項の数 1 (全11頁)

⑮ 発明の名称 計算機システム自動テスト方式

⑯ 特 願 平2-322955

⑰ 出 願 平2(1990)11月28日

⑱ 発 明 者 橋 谷 敏 郎 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア工場内

⑲ 発 明 者 小 玉 育 荘 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア工場内

⑳ 出 願 人 株式会社日立製作所 東京都千代田区神田駿河台4丁目6番地

㉑ 代 理 人 弁理士 小川 勝男 外1名

明 細 書

1. 発明の名称

計算機システム自動テスト方式

2. 特許請求の範囲

1. 複数又は単独の実計算機または仮想計算機から構成される計算機システムで動作するソフトウェアのテストにおいて、複数の独立したテストの実施を同時に実行するためにテスト環境を設定する手続き、テスト資源の使用を設定する手続き、テスト手順を設定する手続きを複数のテスト実施者が独立に指令できる手順と、その解析及びテストの実行の制御を多重に実行する手順を有し、被テスト計算機上のオペレーティングシステム(OS)の発行したコンソールに対する入出力命令を検出する手段と、仮想計算機が発行するコンソールに対する入出力命令要求を検出する手段と、所定の手続きにより被テスト計算機に対するテストジョブを投入する手段と、所定の手続きにより被テスト計算機に対するコマンドを投入する手段と、所定の手続き

により被テスト計算機に対するテストジョブ、コマンドを補正する手段と、テストジョブ及びコマンドの投入を被テスト計算機のOS及び被テスト対象ソフトウェアの状態、テスト経過時間、被テスト計算機の出力情報に対する監視すべき状態に応じて実行すべき動作を制御する手段と、被テスト計算機内で動作するテストプログラムに実行中の計算機システムで出力中のOS及び仮想計算機制御プログラム及びサーバ計算機システムのメッセージ情報を連絡する手段と、実行中の計算機システム内の参照すべきメモリ情報をテストプログラムに連絡する手段と、被テスト計算機のシステムループ、システムウェイトを含む異常状態発生時に異常状態の原因を究明するための資料を自動的に採取する手段と、システムループ、システムウェイトである異常状態発生時に被テスト計算機の最終状態の資料採取後、被テスト計算機での後続のテストを続行するためのテスト実施状態を記憶する手段と、記録したテスト実施状態によりテスト実

特開平4-195436(2)

行のため被テスト計算機での再起動を含むテスト実行のため異常状態発生直前又は当該状態直後に制御状態を補正し、再開始を実行するための制御手段と、上記テスト手続きを格納するファイルより被テスト計算機により出力される出力情報を入力してテスト結果の合否の判定を行う手段と、テスト結果をテストの実行により逐次ファイルに蓄積格納する手段と被テスト計算機間の同期を制御する通信手段と、テスト環境を設定する手続きを格納するファイルと、テスト資源の使用を設定する手続きを格納するファイルと、テスト手順を設定する手続きを格納するファイルと、テスト中に被テスト計算機の再立ち上げ、停止及び使用するテスト資源を指示する手続きを格納するファイルと、一連のテストの実行結果を実行した被テスト計算機全てについて蓄積格納するファイルと、被テスト計算機に投入すべきテストジョブ、コマンドを複数のテスト者は独立に実行するため複数のテスト手続きを格納するファイルと、テスト結果を蓄

ても自動IPL等を行うため自動オペレーション機能を有するものが現れてきている。

従来のテスト自動化の例としてはシステムに処理速度の向上などのレベルアップのような変更を加えた場合、変更の結果が他の部分に影響を与え、誤動作が発生したりする可能性があるため、システムの変更前と変更後にそれぞれ同じコマンドを投入し、結果を比較、照合することが行われている。又TSS(Time Sharing System)等によりコマンドをプロシジャにより自動投入し、同種の効果を上げている事例も存在する。

〔発明が解決しようとする問題点〕

上記従来技術ではテストを実行する計算機システムが実計算機なのか又は仮想計算機なのかの区別をしていない。このため、テストの環境設定や実施できるテストの形態にたいして配慮がなされず、外部仕様の確認を主体とする検査が主体であったが、品質の向上のためには内部仕様にもとづくテストも行う必要があるが、そのためには被テスト環境を仮想計算機に構築して、外部か

蓄積格納するファイルを有し、これらを多重に実行することを特徴とする計算機システムのソフトウェアに対するテスト自動化方式。

3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は大規模ネットワークシステムを構成する計算機システムのソフトウェアをテストする場合、個々のテスト担当者の手動によるテストでは実施工数が莫大となり、総合的なテストの実施が困難なソフトウェア開発時の最終工程である大規模構成下での自動テストの実行に関する。

〔従来技術〕

従来、計算機システムに対するテストにおいては、単独の計算機に対してテストするコマンドを投入したり、ジョブの実行中にコンソールへの応答待ちメッセージに対する応答を行うキーインアウトシミュレーション機能及び本機能を用いたAOM(Automatic Operation Monitor)等によるOSのオペレーション自動化を用いて自動テストを実施している。また仮想計算機システムにおい

ら被テスト環境を制御しながらテストを実施する技術が要求される。又従来技術ではテスト実行時に発生する異常状態に対する検出及び回復手段が配慮されておらず、テスト実行時に異常状態が発生した場合の処置は再び最初からやり直さざるを得なかった。また従来技術では複数の計算機から構成される計算機システムに対するテスト自動化の配慮がなされておらず、複数の計算機例えばディスクの共用、通信ソフトウェアを介したメッセージの送受信を実行したりする等有機的関連を持ったテストに対して適用できない。

本発明の目的は上記の複数の計算機から構成される計算機システムにおいて、個々の被テスト計算機の自動的なテスト実行に加えて複数の被テスト計算機が同時に運転された場合のシステム全体に対するテストをテスト規模に応じたテスト環境の設定と、テストに使用するテスト資源と、テストの実行手続きをそれぞれ複数のテスト担当者が同時に指定して、同時に実行させることを可能とし、テスト実行時での異常状態に対する検出及び

特開平4-195436 (3)

回復手段により同時に実行させることを可能としている。従って長時間に亘る大規模構成の計算機システムに対して自動テストが可能となり今後ますます増大するテスト量に対する対処が可能である。

〔課題を解決するための手段〕

上記目的を達成するため、本発明においては被テスト計算機が実計算機の場合だけではなく、仮想計算機の場合もテスト環境として存在させ、どちらの場合でも、また両者同時の場合でも自動テストを実行させる手段を具備したものである。第1にどの被テスト計算機においても指定したテスト資源とテスト手続きを実行させるために、テスト対象の被テスト計算機の内容に応じてテストを実行するため必要な動作条件を規定するテスト制御手続きと、テストのため使用するテストジョブ、コマンド等のデータの使用を規定するテスト実行資源手続きをそれぞれ独立させた。これにより複数のテスト担当者がそれぞれ独立に手続きを作成することが可能になり、テスト項目生産性の向上

採用している。第5に各被テスト計算機でのテストの実行状況を記録するため、上記各手続きを解釈実行する工程で各手続きの実行状況と、発生異常状態と、手続きに基づく動作の実行記録をファイルに出力して保存する。

〔作用〕

第1に本発明は電子計算機システムのテストは与えられたハードウェア構成を用意して実施される。被テスト対象システムに対しては実計算機を用いてのテストが一般的であるが、それらを仮想計算機を用いてテストすることも可能であり、それらの場合には限られたハードウェア構成上に複数の被テスト計算機を構築することが可能である。

第2に本発明は各被テスト計算機に投入されるテストデータファイル中に蓄積されているテストジョブ、コマンド等はテスト毎に修正されるのではなく、投入時に被テスト計算機で使用可能な形で修正変更するため複数のテストを同時に実行することを可能としている。

第3に本発明は被テスト計算機が実計算機、仮

を達成するものである。第2にこれらの複数のテスト手続きを同時に実行するために、上記のテストを実行するためのテスト動作条件を規定したテスト制御手続きを入力、解釈してテスト動作を開始し、複数の上記テスト実行資源手続きによりテスト資源の使用を可能にし、上記のテスト資源を用いたテスト手順手続きを多重に実行して、テスト実行時での異常状態に対する検出及び回復機構により多重にテストを実行する。

第3にテストの実行によって発生が予想される異常状態に対する監視とそれに対処する動作を可能とするため、各テストジョブ、コマンドの投入に対して監視すべき条件を規定する手続きとそれに基づく監視動作の実行と、異常状態検出時に必要な資料を採取する機構とテストによる出力結果を入力して照合する。

第4に夜間の無人の自動テストを可能とするため、仮想計算機に対する指令とそのコンソールへのメッセージを受け取る機構を有し被テスト計算機のIPLと停止を仮想計算機に連絡する機構を

仮想計算機いずれの場合であってもそれらと独立したテストの手続きを与えることにより、大規模計算機システムのテストにおいて必要なテスト手順の同時実行と実行状況の監視、テスト手続き間の同時制御、テスト進捗状況の記録、及び被テスト計算機の立ち上げ、停止を自動的に実行することにより、複数のテスト作業を並行して実施可能であり、期待するテストが容易に実施されることになる。

第4に本発明はテスト実施状態を外部記憶装置に引き継いでおり、被テスト計算機のIPLを何度も途中に含むテストについて実行することが出来る。同時にテストの実行時での異常状態に対する検出及び回復を行うことによりテストの自動継続や再実行を自動実行でき、テスト要員を最小限にすることが出来る。

〔実施例〕

以下、本発明を図面を参照しつつ説明する。

第1図は本実施例の全体構成図である。1は自動テストが実行される計算機システムである。2

特開平4-195436 (4)

は仮想計算機を制御する制御プログラムである。(以下、単にVMCPという。)3はVMCPと連動してその動作を制御する仮想計算機であり、VMCPの動作を制御する。(以下OSTDと略称する。)4-1, 4-2はテスト対象のソフトウェアのソフトウェアを仮想計算機環境で動作させる場合の仮想計算機システムである。(以下VMと略称する。)10はテスト対象のソフトウェアを実計算機環境で動作させる場合の実計算機システムである。(以下RMと略称する。)9は2, 3, 4-1, 4-2および10から構成されるテスト対象のソフトウェアを動作させる複合計算機システムである。5-a, 5-b, 5-c, 5-dはそれぞれOSTD, VM, 又はRM上においてテストの実行を制御する本発明システムである。(以下ATSと略称する。)6はVMにおける仮想コンソールディスプレイ、7はVMにおける仮想プリンタである。11はRMにおける実コンソールディスプレイ、12はRMにおける実出力プリンタである。8はOSTDを動作させるための

って開始される。5-a, 5-b, 5-c, 5-dはそれぞれ動作が独立に開始されるが、ファイル13の手続きを入力、解釈することにより、テストされる計算機環境がどのような構成で動作しているか識別し、それぞれ3, 4-1, 4-2及び10で動作するATS間の接続経路を確立し、システムの同期合わせを行なった後その処理を開始する。

5-aは13に格納されているテスト環境定義手続きに指示されている使用環境の定義とファイル8に格納されている仮想計算機動作手続きにもとづいて、4-1あるいは4-2のVMの実行の開始をOSTDを経由してVMCPに連絡する。5-b, 5-c, そして5-dは上記起動の後ファイル14のファイルよりテストに用いる資源を定義する手続きを入力、解析してそれぞれVM又はRM上において実行すべきテスト即ちオンラインシステム、TSSジョブ、バッチジョブ群等に対応したテストの論理的単位であるテストプロセスを必要な数だけ生成して、それぞれのテストプ

手続きを格納するファイルである。13はテスト環境を定義する手続きを格納するファイル、14はテストに使用する資源を定義する手続きを格納するファイル、15はテストの手順、内容を定義する手続きを格納するファイル、16はテスト実行状況を逐次蓄積するファイル、17はテストの実行にともなって発生するRM又はVMにおける6または11に対する出力情報を逐次蓄積するファイル、18は7または12に対するテストの実行結果を逐次蓄積するファイル、19は被テスト計算機に投入するテストジョブ、テストコマンド、テストデータが格納されているファイル、20は被テスト計算機が異常状態となった場合に5-a, 5-b, 5-cがテストの続行のためその実行状態を記録するファイルである。

テスト対象のソフトウェアのテストは3の開始とそれに連動する5-aの起動、及び5-aより3に対する指令による4-1, あるいは4-2の起動と、それに連動する5-b及び5-cの起動、10においてはその開始直後の5-dの起動によ

ロセスに対応するテスト手順を格納するファイル15よりテスト手続きを入力、解析し、ファイル19に格納されているテスト資源より必要なテストデータをそれぞれVM又はRMへ投入していく。テストの実行に伴い発生する事象はファイル16にテストログ情報として蓄積され、テスト実行後に別プログラムによって解析される。5-b, 5-cそして5-dはVMでは6に対して、RMでは11に対して発行されるコンソール入出力情報を発生順にファイル17に蓄積格納する。同時にVMでは7に対して、RMでは12に対して出力される出力結果を、第2回において説明する工程により選別し、ファイル18に蓄積格納する。

第2回は3, 4-1, 4-2, 及び10において実行される5-a, 5-b, 5-c, 5-dの制御機構を示している。ATSの動作はシステム全体制御部(ATMC)100によって開始される。ATMCは動作のために必要なファイルの割当てとオープン、使用メモリ資源の確保と初期設定を行った後、テスト環境定義手続き入力解析部

特開平4-195436(5)

(ATCP) 110、テスト資源定義手続き入力解析部(A T E P) 120を順次動作させる。また3で動作するATSの場合テスト実行に先立って4-1、4-2の起動を仮想計算機連絡制御部(ATVM) 153より3へ連絡することにより実行する。

ATCPはファイル13よりテスト環境定義手続きを入力、解析して、制御部ATMCによって確保された制御テーブルにその解析結果を格納する。同時に指定された個数のテストプロセスを生成し、その処理完了後ATMCへ制御を戻す。ATMCは制御部ATEPに対して、上記ATCPによる解析結果をその入力として制御を渡す。ATEPにおいては使用テスト手順手続きファイル情報により当該ファイルの動的な割当てとオープン、使用するテストデータファイルの情報により当該ファイルの動的な割当てとオープン、動作条件に対する手続きからの情報をもとに処理を実行しテスト資源を使用可能にし、ATMCに制御を渡す。

制御部ATCCは生成された複数のテストプロセスに対する処理を多重に実行することにより複数のテストの多重実行を可能としている。制御部ATCCはテスト実行手順手続き入力解析部(ATSP) 130に対してテスト手順の手続きの入力、解析を指令する。ATSPはテスト手順手続きを第1図のファイル15より入力し、その解析を行い、その結果を第3図で説明されるキューとして登録する。本処理により生成されるキューREQQ200はシステムへの投入に必要なテストジョブのファイル中のメンバ名称、応答待ちメッセージ発生時の応答条件手続きのメンバ名称、ジョブ実行の結果リスト解析手続きのメンバ名称、最大実行許容時間、動作条件、又は投入すべき被テスト計算機に対するコマンド、又はVMの制御コマンド、又は発生が予想される状態について監視すべき情報を指示する項目を含む。

ATCCはATSPの処理完了の報告後テスト資源管理制御部(ATEX) 140にキューREQQをパラメタとして渡し、動作を指令する。

本システムの被テスト計算機の動作を監視するモニタ部は、メッセージ入出力について監視し、登録されたメッセージが発生した場合にATMC連絡を行うメッセージ入出力モニタ部(ATMS) 150、発生時の監視を指定された事象が被テスト計算機内に発生した場合ATCCに連絡を行う被テスト計算機状態モニタ部(ATMN) 151、指定された経過時間を監視しATCCに時間の経過を報告する経過時間モニタ部(ATTM) 152よりなる。

上記処理完了後、ATMCはコンソール入出力を処理するメッセージ入出力モニタ部(ATMS) 150、被テスト計算機状態モニタ部(ATMN) 151、被テスト計算機時間モニタ部(ATTM) 152を起動した後、テストプロセスの実行を制御するテストプロセス制御部(ATCC) 160を起動するとともに、コンソール入出力の監視処理及びテスト環境定義手続きに定められた条件によりテスト実行時間の監視を制御部(ATTM) 152に対して指令する。

ATEXはキューREQQの情報により、テスト実行のための制御情報を保持するキューJOBQ210、コマンド情報を保持するキューCMDQ220、メッセージ情報を保持するキューMSGQ250または経過時間の検出のためのキューTIMQ230とともに、監視すべき条件の成立時の動作を規定するACTQ220を生成して監視条件手続き入力、解析部(ATCN) 141に制御を渡し、テストプロセス実行中に発生が予想される応答付きメッセージに対する応答の内容の定義、指示の事象即ち発生するメッセージ、経過時間、被テスト計算機のメモリの内容、資源(CPU使用率、スプールファイル使用率、入出力使用率等)の監視事象とそれに対する動作内容即ちコマンドの投入、指定したジョブの投入、実行、メモリダンプの採取、指定テストプロセスの停止、及び当該被テスト計算機即時停止指示をATSに登録する。これらのテスト実行に関連する情報は全て第3図においけるキューに変換され、それぞれテストプロセスの実行に伴ってモニタ部によ

特開平4-195436 (6)

て各キューを解析実行することによって所定の動作が行われる。ATCCはモニタ部ATMS, ATMN, ATTMからの連絡を受けた場合、第3図に示されたキューの関連情報にもとづき対応する処理を実行する。

以上説明した一連の動作がすべてのテスト手順を処理するまでテストプロセスごとに多重に繰り返される。

メモリダンプの採取、被テスト計算機の停止指令は事象発生時、ATMCより、被テスト計算機がVMの場合には仮想計算機に対する指令を実行する仮想計算機連絡制御部(ATVM)153を経由して3に対して連絡され、RMの場合には直接の指令をRMに対して連絡することによってその処理が行われる。

第3図は第2図により説明した制御手順で処理される情報について、処理に使用される内部情報としての関連を表したものである。

REQQ200はテスト手順の情報を保持するキューであり、ジョブの実行、又はコマンドの投

ACTQ220はキュー210, 230, 240, 260よりポイントされ、メッセージの発生、メモリ情報、実行時間、システムの状態等の監視事象が発生した場合の処理を規定する情報を保持するキューであり、コマンドの投入、応答付きメッセージに対する応答、他のシステムで動作中のATSへの連絡、あらかじめ用意されたジョブの投入、ダンプの採取、システムの停止、メッセージの出力、ジョブのキャンセル等の動作を規定する。

JOBQ230はシステムにおいて時間の経過の検出に用いるキューであり、キュー230を確保し時間間隔をキューへ指定すると、指定された時間に達した時に連絡要求元へ時間の経過が連絡される。これらはキュー200, 210の他システムの処理で様々な用途に用いられる。

CMDDQ240はキュー200, 220よりポイントされ、投入すべきコマンドを保持するキューである。

MSGQ250はキュー200, 210よりポ

入、又は指定時間の経過の待ち、又はメッセージの発生、メモリ情報、実行時間、システムの状態等の監視、又はテスト手順の投入終了のテスト手順種別と、テスト手順種別に従って作成される関連するキュー210, 230, 240, 250へのポインタを保持している。テスト手順の種別がジョブの実行の場合には、ジョブのファイル中でのメンバ名、メッセージの発生、メモリ情報、実行時間、システムの状態等の監視すべき事象とそれに対する処理を記述する手順のファイル中でのメンバ名、ジョブ終了時にジョブ出力結果について検証すべき条件を記述する手順のファイル中でのメンバ名を保持する。キュー200はテスト手順の解析時確保され、テスト手順処理が終了すると他の関連するキューとともに削減する。

JOBQ210はキュー200のテスト手順種別がジョブの実行の場合に確保され、ジョブの実行に対してテスト手順で指示された条件、及びジョブ実行中において、開始時間、終了時間、使用したシステムの資源の記録を保持する。

イントされ、監視すべきメッセージの内容を保持するキューである。メッセージの発生の度にキュー250に登録されているメッセージの内容と一致するか否かが判定され、一致した場合には登録した連絡要求元へその発生が連絡される。

第4図はテストプログラム間又はATSとテストプログラムの間で行われる情報の送受信について示している。テストプログラム310とテストプログラム320間で同期を取ったり、互いに情報を送受信したい場合、テストプログラムの動作中、メッセージ情報やメモリ情報を参照したい場合、その処理の種類を初期設定でATSへ連絡した後、必要な時点でATSに処理を要求する。ATSは被テスト計算機で動作中のテストプログラムより上記処理要求があった場合には、第2図で説明した制御機構で要求の情報の送受信を行う。要求の種類が他被テスト計算機に対するものの場合、第5図で示される連絡機構を用いて処理を行う。要求の種類が被テスト計算機のコンソール情報の場合にはATSのモニタ部により作動中のすべ

特開平4-195436(7)

ての被テスト計算機のコンソール330の入出力情報をテストプログラムの要求があり次第渡す。要求の種類が被テスト計算機のメモリ情報の場合はすべての被テスト計算機についてATSのモニタ部により一定時間間隔で監視しているメモリの内容340がテストプログラムの要求があり次第渡す。VMに対してはVMCP又はOSTDのコマンドの形式で命令を発行する。

ATCCはテスト手順311, 312の手続きを1個づつ処理するが、テストジョブ、コマンドの投入は、テストジョブ、コマンド投入部210に指令をだして実行する。又テストジョブの終了時にはテスト結果入力照合判定部(AOT)170により、結果リスト中の特定文字列の探索により所定の実行結果を得たか否かを判定する。またこれらの実行についてその更新交信の記録がファイル16に逐次蓄積格納される。

第5図は複数のシステム間での情報の連絡を行うための通信手順の関連を表したものである。各被テスト計算機で動作するシステム440, 45

0は、仮想計算機で動作する場合には共通なメモリ400を、実計算機で動作する場合には共通な外部記憶ファイル410を使用し、400又は410を一定時間の間隔で参照して他のATSからの連絡がないかどうかの検出処理を行う。所定の被テスト計算機で動作するATSが他のATSへ情報を通信する場合には連絡情報を上記共通連絡領域400、又は共通連絡ファイル410へ出力することのより実行される。第5図においてのATS間の連絡は、3への指令によりVMCP2が特定の共通なメモリ400を書替えることによりなされる。互いに二つの被テスト計算機計算機間で連絡が有ったかどうかの有無及び相手計算機が作動中か否かの監視動作は以上の機能を使用してATSがその状態を把握し、テストプロセスの実行を進めていく。それぞれのATSは一定時間の間隔で400又は410を参照することにより被テスト計算機システムでどれだけのシステムが動作中か、又被テスト計算機のシステムダウン等が発生していないかどかを検証して、特定の被テ

スト計算機の異常状態が発生していることを検出した場合、最初に検出したATSがそれに対応する回復動作を実行する。

第6図はテストデータファイル19に格納されているテストデータに対して、テスト環境に合わせた変更を行う関連を表したものである。テストデータファイル中のテストデータは例えば使用するディスクの場合には作成時の想定テスト環境500あるいは510を前提として作成されているが、想定テスト環境が実際のテスト環境と異なっている520あるいは530である場合、その旨テスト環境及びテスト手順手続き中に指定することにより動作させるため、以下の一部情報の補正処理を行う。テストデータ入出力部(ATIO)142で入力されたテストジョブ、コマンドはテスト環境定義手続き入力解析部(ATCP)110により入力解析される環境定義手続きにより作成されるシステム環境状態を保持するテーブル570、及びテスト資源定義手続き入力解析部(ATSP)130で入力され解析するテスト実行手

続き中であらかじめ指定されて作成された変換情報を保持するテーブル580にもとづいて142によりテストデータ入力時、データの一部分が540から550へと変更補正され、被テスト計算機で動作可能な形に変換された後、142によって被テスト計算機に投入される。

第7図はファイル13, 14, 15中に格納されているテスト環境、テスト資源、及びテスト手順の続き間の関連、及び被テスト計算機で異常状態によりシステムの実行が不可となった場合にOSの再IPL(Initial Program Load)を含むテストの再実行を行う場合のテスト環境、テスト資源手続き、テスト手続き、テスト実施状況を引き継ぐ場合の関連について表したものである。

テスト環境、テスト資源、テスト手続きはそれぞれ640, 650そして660に分かれておりそれぞれ対応する情報を格納して手続きに従って関連づけられている。640, 650, そして660はファイル600に同時に記録される。ファイル600はテストの実行状況を逐次蓄積して格

特開平4-195436(8)

納部610及びそれぞれテストプロセスに対応して格納部620、格納部630を保持する。格納部620、格納部630はそれぞれ指定された個数分生成される。格納部610はテスト環境手続きの解析の結果による情報640を格納するとともに、現在どのテスト環境を使用してテストが実行されているのかを保持する。格納部620はテスト資源定義手続きの解析の結果による情報650を格納するとともに、現在どのテスト資源を使用して当該テストプロセスがどこまで実行されているのかを保持する。格納部630は実行中のテストプロセスにおいて実行中のテストの手順、内容を定義する手続きの解析の結果による情報660を格納するとともに、現在どのテスト手順まで実行されているのかを保持する。第7図におけるファイル600においては2つのテストプロセスが存在して1つ目のテストプロセスはテスト手順B1を2つ目のテストプロセスはテスト手順Bnを実行中であり、テスト手順はそれぞれC1、Cnを処理中であることを表している。

指示をそれぞれの制御部に指示する。

テストの実行に伴うファイル600の更新は以下のごとく行われる。テスト環境定義手続き入力解析部(ATCP)110がファイル14よりテストに用いる資源を定義する手続きを入力、解釈する時、同時に当該手続きの先頭からの相対位置B1やBnを順次ファイル600に記録する。また各テストプロセスに対応してテスト手順を格納するファイル15より手続きを入力、解釈する際、同時に当該手順の先頭からの相対位置C1やCnを順次ファイル600に記録する。

複数の被テスト計算機間で動作中のATSは各被テスト計算機で互いにファイル600をその起動時アクセスして、テスト状態がどの被テスト計算機からも同等に参照することが可能である。互いに二つの被テスト計算機間で連絡があったかどうかの有無及び相手計算機が作動中か否かの監視動作は第5図で示された機構を使用して実行され、各被テスト計算機でその状態が把握される。異常状態の発生またはシステムダウン等によ

ファイル600の初期設定4-1、4-2の開始直後最初にアクセスしたATSによって行われるが、当該ファイルの状態によって、正常な立ち上げなのか、再開始立ち上げなのかを判別して、自システムの処理形態を決定する。処理形態を決定はテスト環境定義手続き入力解析部(ATCP)110により、環境定義の手続き解析時指定された手続き中にIPL手続きを含む再開始手続きが何回含まれているかカウントし、最終状態を示すフラグが実行中であつた再開始カウンタが初期値でなければ再開始立ち上げであると判断する。異常状態発生等の再立ち上げの場合はテスト資源定義手続き入力解析部(ATEP)120において、格納部620及び格納部630に格納されている最終の状態まで手続きの解析をスキップして開始処理が行われる。立ち上げ時のシステムの運転の指令内容によっては上記の状態を無視して最初の手続きから処理を行ったり、最終の状態の次の手続きから処理を行ったりすることも可能である。ATMC100はこれらの動作を制御し、必要な

連絡動作の停止の検出は異常状態発生当該被テスト計算機のシステム以外で動作するATSで検出され、それらに対してあらかじめ指定された手順にもとづいて再開始の操作が当該被テスト計算機に対して行われ、引き続き異常状態の検出された被テスト計算機でのテストが続行される。

〔発明の効果〕

本発明は、異常説明したように構成されているので以下に記載されるような効果を奏する。

第1にテスト環境に関する手続き、テスト資源に関する手続き、テスト手順に関する手続きをそれぞれ独立に複数与えることによりテスト条件の設定が従来の単一的手続きを与える手法にたいして、同時に複数のテスト担当者で設定したテストを同時実行できる。

第2に仮想計算機システムとインタフェースを保持してシステムの動作が可能であり、テストの実行に伴う被テスト計算機の状態監視、計算機間の同時制御、レジスタ情報、メモリ情報の必要時点での採取、被テスト計算機の停止等の従来手法

特開平4-195436(9)

では実現が困難であった複数計算機でのテストをも可能とし、テストの効率向上を実現できる。

第3に上記の各手続きを一連の手続きとして一括して処理する場合、互いに独立な要素である手続きの内容が分かれず、保守効率が低い。従ってテスト環境に関する手続き中で、個々のテストプロセスで使用するテスト資源の定義を手続きとして規定し、被テスト計算機に対する環境、動作とそれにより行われるテストプロセスを階層化した。又それぞれのテスト手順はテストプロセスの定義手続き中で使用を指定することにより、同様に独立性を保持させている。この階層構造により、従来に比べてメンテナンスの効率面からでなく、テストの実行に関する手続きの表現がよりわかりやすくなりテスト準備作業工数を低減できる。

第4に一度作成したテストデータはテスト環境が変化しても、データの一部補正を行うため、テスト環境が変化する度に修正することが不要であり、テストデータの格納スペースを低減できる。同時にテスト環境設定についても共通化を図るこ

とにより設備の投資工数を削減することができる。

第5に異常状態の発生によるテスト中断を再開機能により回避することが出来、それに伴うテスト時間のロスを低減することができる。

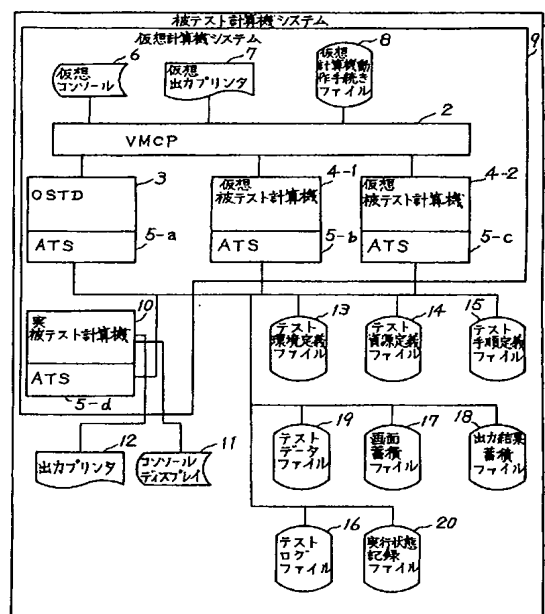
4. 図面の簡単な説明

第1図は本発明の実施例の構成を示す機能ブロック図、第2図は本発明の制御機構を説明するシステムの制御部間の関連図、第3図はテスト動作環境定義手続き、生成されるテストプロセスで使用するテスト資源定義手続きと使用するテスト手順手続き、及びそれらに対する動作条件の関連図、第4図はテストプログラム間又はATSとテストプログラムの間で行われる情報の送受信についての関連図、第5図は複数の被テスト計算機システム間での情報の連絡を行うための円環についての関連図、第6図はテストデータファイルに格納されているテストデータに対して、テスト環境に合わせた変更の実行について関連図、第7図はテスト環境、テスト資源、及びテスト手順の手続き間の関連、及び被テスト計算機で異常状態によりシ

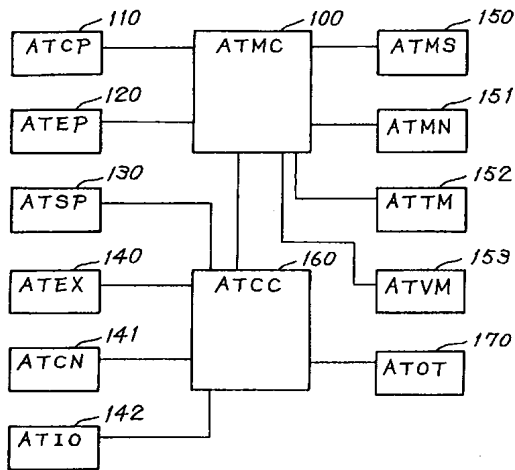
ステムの実行が不可となった場合の回復動作についての関連について表した図である。

1…被テスト計算機、2…仮想計算機制御プログラム(VMCP)、4-1、4-2…仮想計算機(VM)、6…仮想コンソール、9…仮想計算システム、10…実計算機、12…実コンソールディスプレイ、13…テスト環境定義ファイル、14…テスト資源定義ファイル、15…テスト手順定義ファイル、16…テストログファイル、17…画面蓄積ファイル、18…出力結果蓄積ファイル、19…テストデータファイル、20…実行状態記録ファイル。

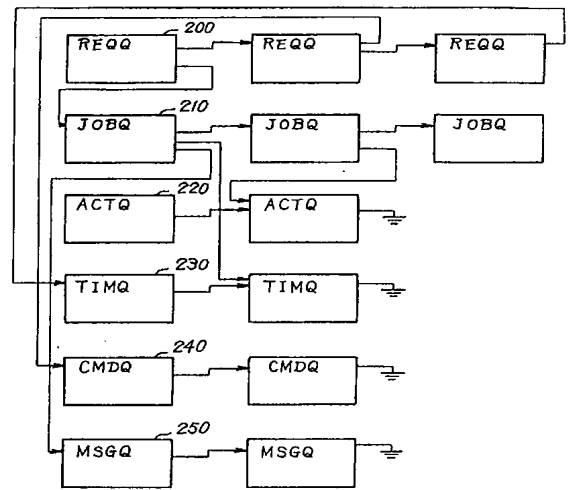
第1図



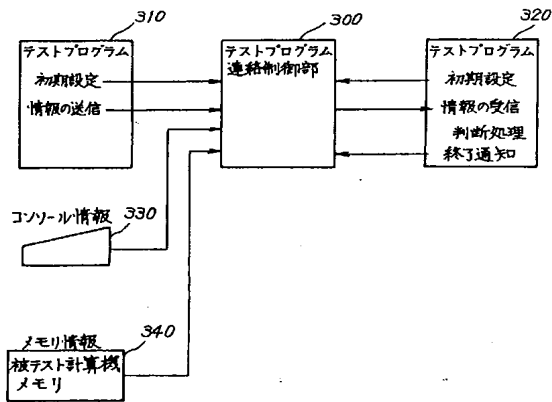
第 2 図



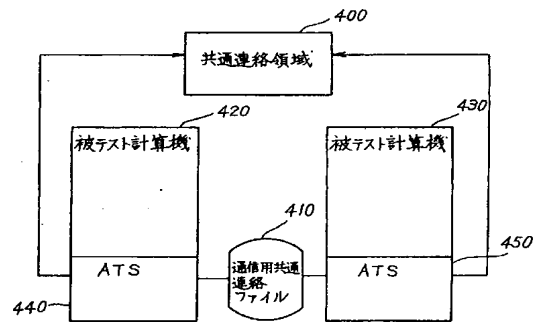
第 3 図



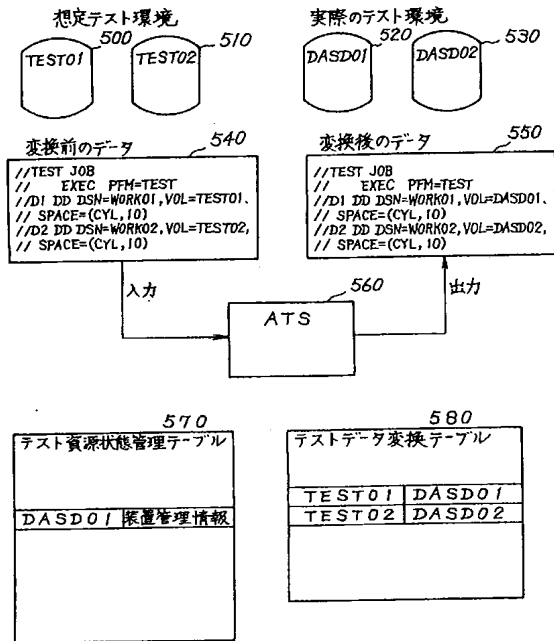
第 4 図



第 5 図



第 6 図



第 7 図

